## IN THE CLAIMS:

Please, cancel claim 1 and add new claims 2-21. All claims and their current status are listed below.

1. (Cancel)

2. (New) A method of managing memory in a microprocessor based system comprising the steps of:

storing packet data in a physical address space of a memory, the physical address space having a size;

generating a logical index of the physical address space, the logical index having a logical address space at least twice the size of the physical address space;

mapping in the logical index each physical address to two logical addresses; the two logical addresses spaced apart in the logical index by at least the size of the physical address space; and

generating an access pointer in the logical index to provide bi-directional logical access to the logical address space.

3. (New) The method of claim 2 further comprising:

pointing the access pointer to a middle logical address of the logical address space;

writing in the direction of increasing logical addresses a data packet having a first header of a first number of bits and a last bit;

pointing the access pointer to the logical address of the last bit of the first header of the data packet; and

writing in the direction of decreasing logical address a second header having a second number of bits including a last bit, wherein the second number of bits is different

than the first number of bits, and wherein a first bit written corresponds to the last bit of the second header.

4. (New) The method of claim 2 further comprising:

generating one or more cloned logical indexes of the physical address space, the cloned logical indexes having each a cloned logical address space equal to the logical address space; and

indicating in each physical memory address that it is being mapped in two or more logical indexes in response to generating the one or more cloned logical indexes.

5. (New) The method of claim 4 further comprising, in response to an instruction to store data in a first physical address being identified by a first logical address and indicating that it is being mapped in two or more logical indexes:

copying the contents of the first physical address to a second physical address; and

modifying the first logical address to map the second physical address.

6. (New) The method of claim 2, wherein the memory comprises two or more memory devices.

7. (New) The method of claim 6, wherein the two or more memory devices include one of the group consisting of a PRAM, a main memory, and an off-chip data memory.

8. (New) A method of buffering packet data in a micro-processor based packet data communications system, comprising the steps of:

generating a byte-addressable netbuf having a list of logical memory addresses, each logical memory address mapping a physical memory address corresponding to a fixed-size physical memory location in a memory device, wherein each fixed-size

memory location is mapped by two logical memory addresses in a byte-addressable netbuf;

generating a set of pointers indicating the first logical memory address, the last logical memory address, and the logical memory address to be accessed of the byte-addressable netbuf; and

managing the byte-addressable netbufs by modifying the set of pointers to process packet data in the fixed-size physical memory locations, wherein modifying includes identifying in the set of pointers the logical memory addresses in the byte-addressable netbuf that map to the physical memory addresses of the fixed-size memory locations.

9. (New) The method of claim 8 wherein the managing comprises duplicating an original byte-addressable netbuf by generating a cloned netbuf comprising a cloned list of logical memory addresses mapping the same physical memory addresses of the same fixed-size physical memory locations mapped in the original byte-addressable netbuf.

10. (New) The method of claim 9 wherein the modifying the set of pointers comprises, in response to a command to store data in a first fixed-size physical memory location mapped in the original byte-addressable netbuf and in a first logical memory address of the cloned netbuf:

copying the contents of the first fixed-size physical memory location to a second fixed-size physical memory location;

changing the first logical memory address to map the second fixed-size physical memory location; and

storing the data in the second fixed-size physical memory location.

11. (New The method of claim 8 wherein the set of pointers includes a start pointer indicating a first logical memory address, an end pointer indicating a last logical memory address, and an access pointer indicating a logical memory address between the first logical memory address and the last logical memory address.

12. (New) The method of claim 11 wherein the modifying comprises:

storing in a set of fixed-size physical memory locations in an increasing logical address direction starting at a first fixed-size memory location mapped by the logical memory address indicated in the access pointer a data packet having a payload data section and a header data section, wherein the header data section comprises a first header data corresponding to a first communications protocol;

indicating a second logical memory address in the access pointer corresponding to a last fixed-size physical memory location containing the first header data; and

storing in a decreasing address direction a second header data corresponding to a second communications protocol, wherein the second header data includes a different number of bits than the first header data.

13. (New) The method of claim 8 wherein at least one of the fixed-size memory locations is a non-byte-addressable memory location and wherein an application executing in the micro-processor based system supplies a command to process packet data in the non-byte-addressable memory location by referring to a corresponding byte-addressable logical address in the list of the byte-addressable netbuf.

14. (New) In a memory-constrained, microprocessor-based system, a method of implementing multiple packet data communications protocols having protocol-specific headers, the method comprising the steps of:

allocating physical memory locations in one or more memory devices to fixed-size netpages for storage of packet data, the fixed-size netpages having a netpage identifier and a set of physical memory addresses;

mapping each fixed-size netpage to two logical memory addresses; and

composing a netbuf including an index list having the two logical addresses of each fixed-size netpage within a set of fixed-size netpages, wherein the index list of the netbuf comprises a logical memory space at least twice the size of a physical memory space corresponding to the set of fixed-size netpages.

15. (New) The method of claim 14 further comprising:

storing in the set of fixed-size netpages in a first logical address direction, a data packet having a first protocol-specific header including a last header bit; and

replacing the first protocol-specific header with a second protocol-specific header by storing in a second direction opposite to the first direction the second protocol-specific header starting at the fixed-size netpage memory address containing the last bit of the first protocol specific header.

16. (New) The method of claim 14 further comprising:

storing in the set of fixed-size netpages in a first logical address direction, a data packet having a first protocol-specific header including a last header bit; and

composing a cloned netbuf including a copy of the index list;

copying the first protocol-specific header to a second set of fixed-size netpages;

modifying the copy of the index list in the cloned netbuf by changing the logical addresses mapping fixed-size netpages of the first set of fixed-size netpages

containing the first protocol-specific header to map the second set of fixed-size netpages; and

replacing the first protocol-specific header in the second set of fixed-size netpages with a second protocol-specific header by storing in a second direction opposite to the first direction the second protocol-specific header.

17. (New) The method of claim 14 wherein two or more fixed-size netpages are in noncontiguous memory blocks.

18. (New) The method of claim 14 wherein the netpage identifiers of each fixed-size netpage indicate if the netpage is available for data storage or already allocated.

19. (New) The method of claim 18 wherein the netpage identifier indicates that the fixed-size netpage is available for data storage by indicating a physical memory address of a next available fixed-size netpage, and wherein the netpage identifier indicates that the fixed-size netpage is already allocated by indicating the number of index lists mapping the fixed-size netpage.

20. (New) A gateway system for bridging packet data networks using two or more communications protocols having protocol-specific headers, the gateway system comprising:

a micro-processor;

a secondary memory device electrically coupled to the micro-processor for data storage; and

a main memory device electrically coupled to the micro-processor for storage of instructions and data, wherein the instructions when executed by the micro-processor generate a logical index that maps two logical memory addresses to

each of a set of physical memory locations, each physical memory location located in the main memory device or in the secondary memory device, and generates a set of pointers for receiving logical memory addresses that enable applications in the gateway to bi-directionally access and replace protocol-specific headers in the physical memory locations to transfer data from a first data network using one communications protocol to a second data network using a second communications protocol.

21. (New) An apparatus for buffering packet data in a micro-processor based packet data communications system, comprising:

means for generating a byte-addressable netbuf having a list of logical memory addresses, each mapping a fixed-size physical memory location in a main memory device and in a secondary memory device, each fixed-size memory location having a physical memory address being mapped by two logical memory addresses;

means for generating a set of pointers indicating the first logical memory address, the last logical memory address, and the logical memory address to be accessed of the byte-addressable netbuf; and

means for managing the byte-addressable netbufs by modifying the set of pointers to process packet data in the fixed-size physical memory locations by indicating the logical memory addresses in the byte-addressable netbuf that map to the physical memory addresses of the fixed-size memory location.